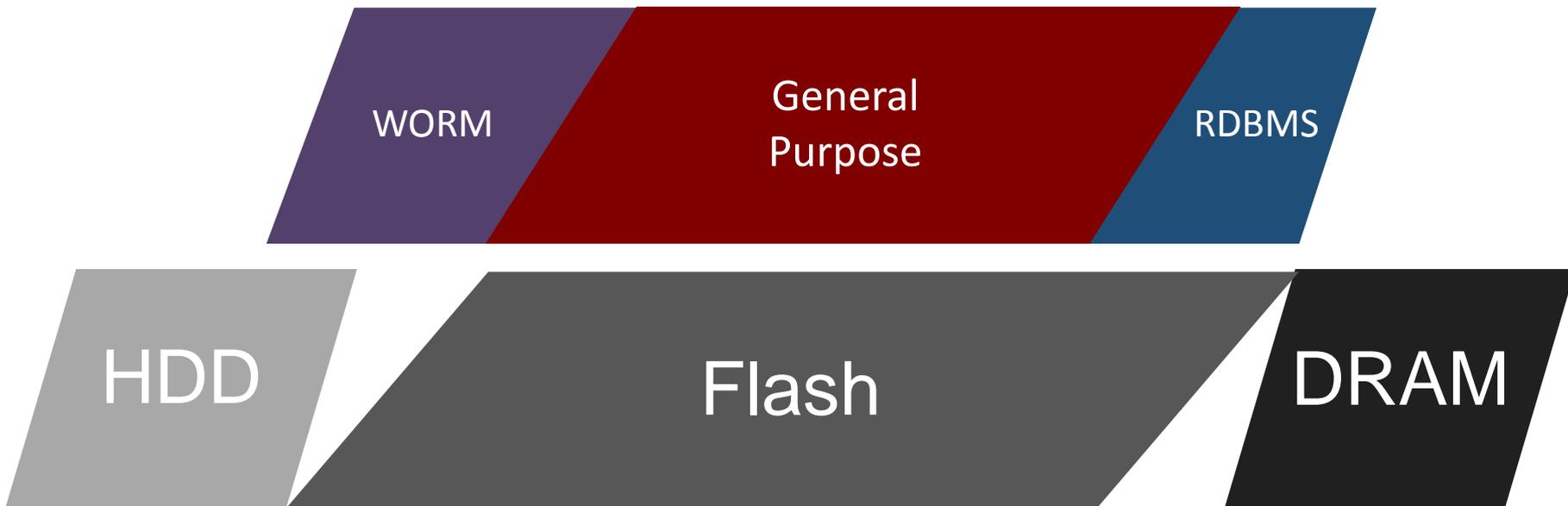


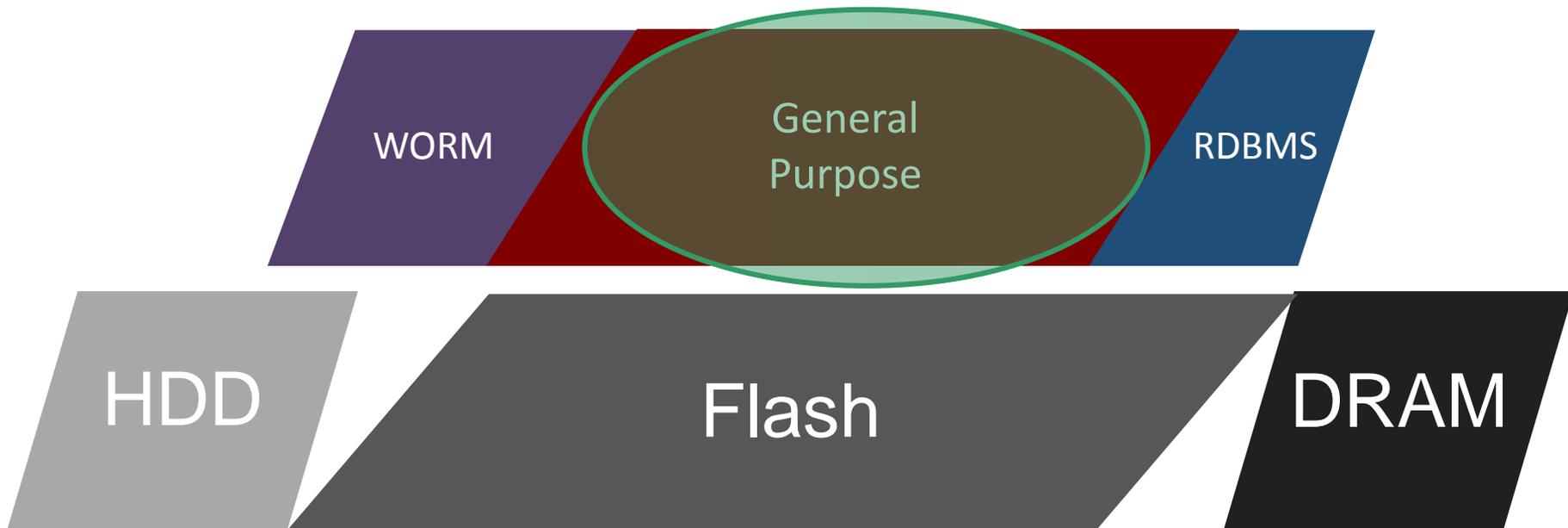
# Replacing the FTL with Cooperative Flash Management

Mike Jadon  
Radian Memory Systems  
[www.radianmemory.com](http://www.radianmemory.com)

# Data Center Primary Storage



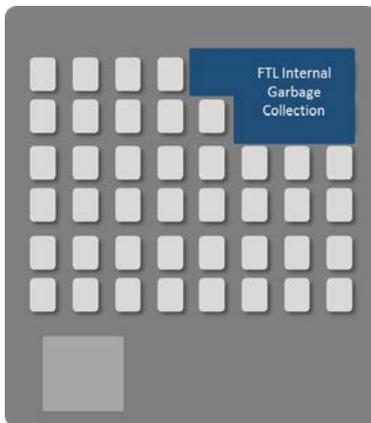
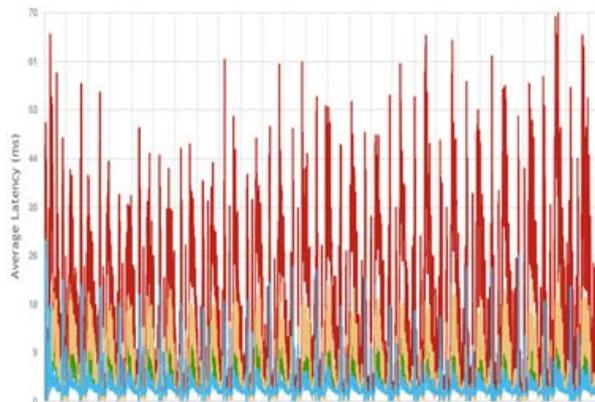
# Data Center Primary Storage





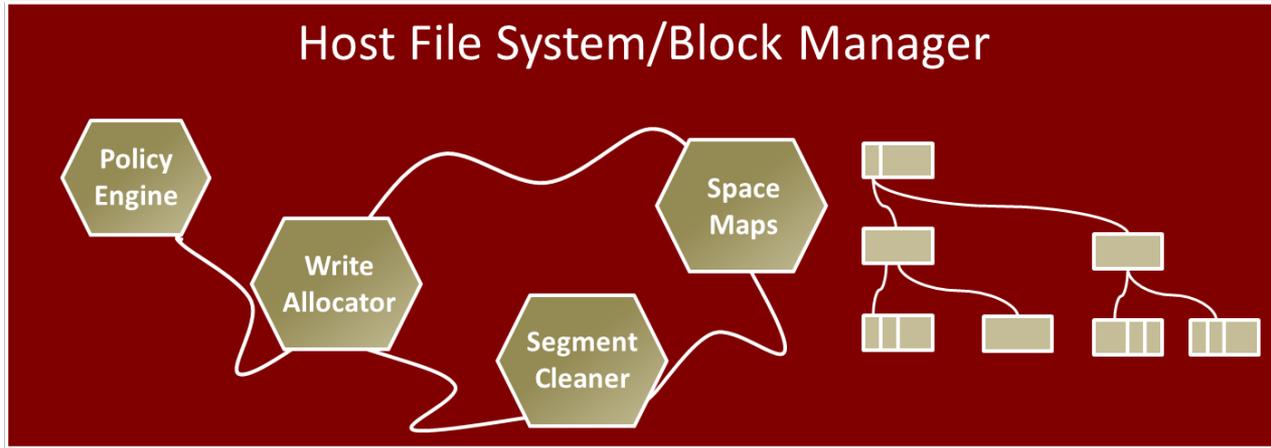
Flash SSD

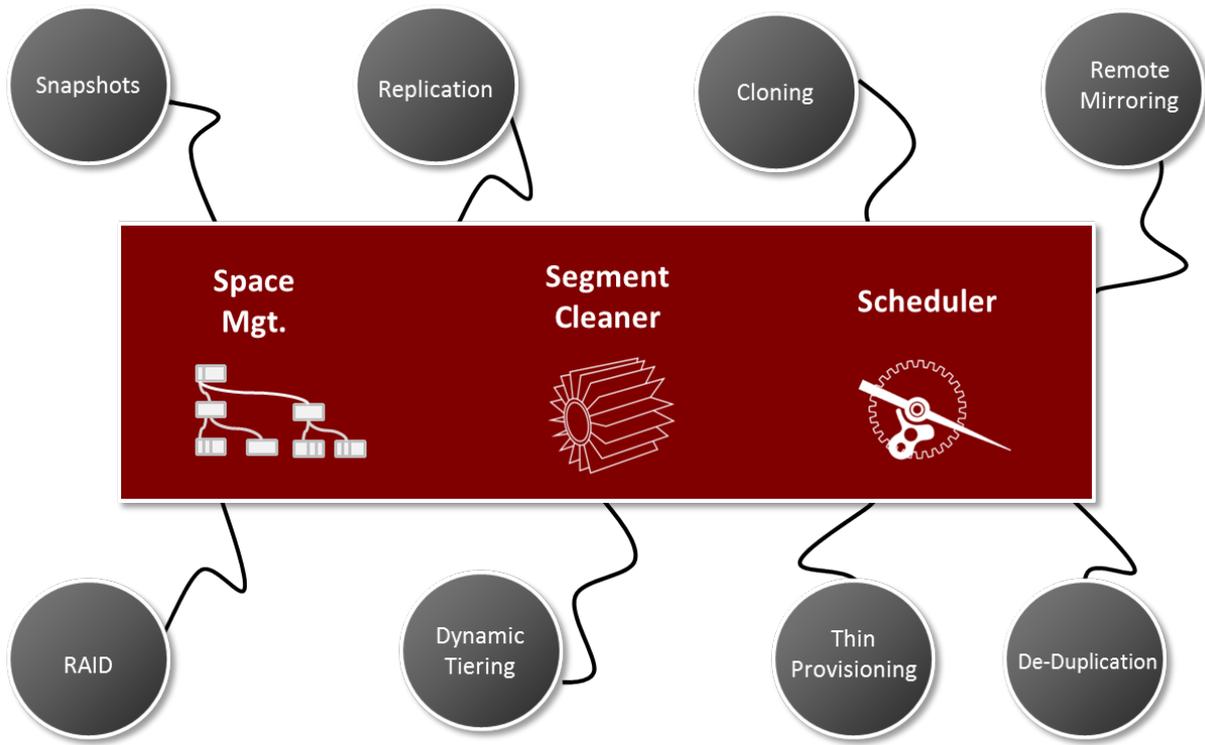
- Unpredictable Latency Spikes (QoS)
- Cost
- Endurance (Wear Out/TCO)

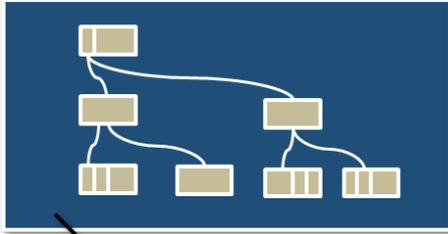


# Advanced Systems

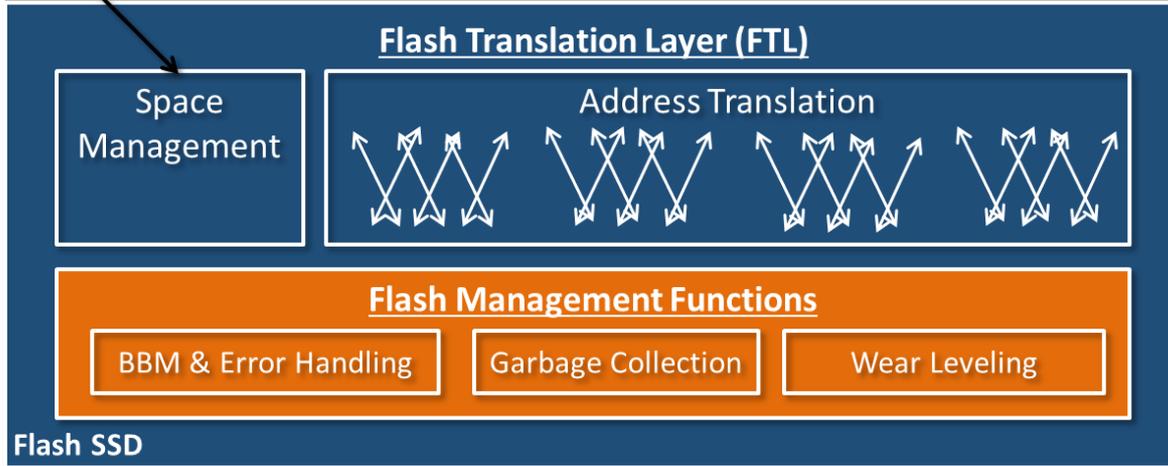
- Purpose-Built storage systems
- All-Flash Arrays
- Hybrid SSD/HDD
- Advanced local file systems
- Software-Defined Storage architectures
- Hyperconverged systems
- Object/Key Value Stores
- Certain In-Memory Data Base applications







- Backwards Compatible
- Simple Contract



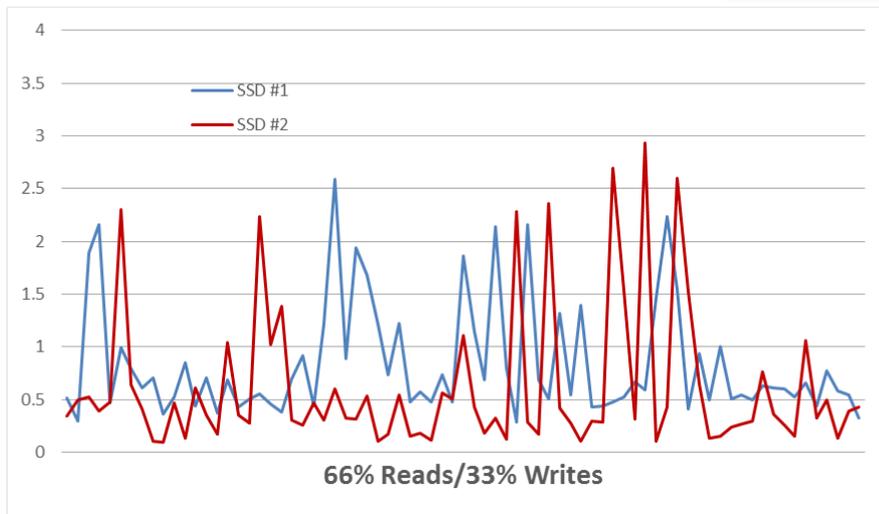
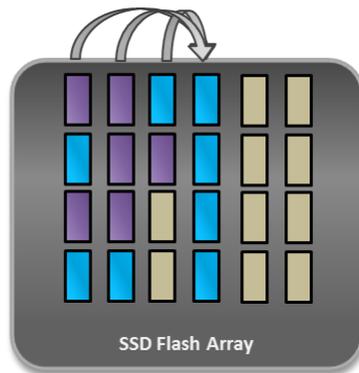
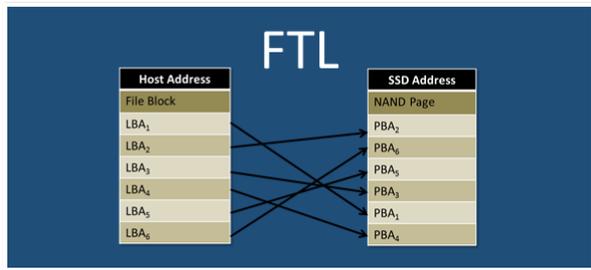
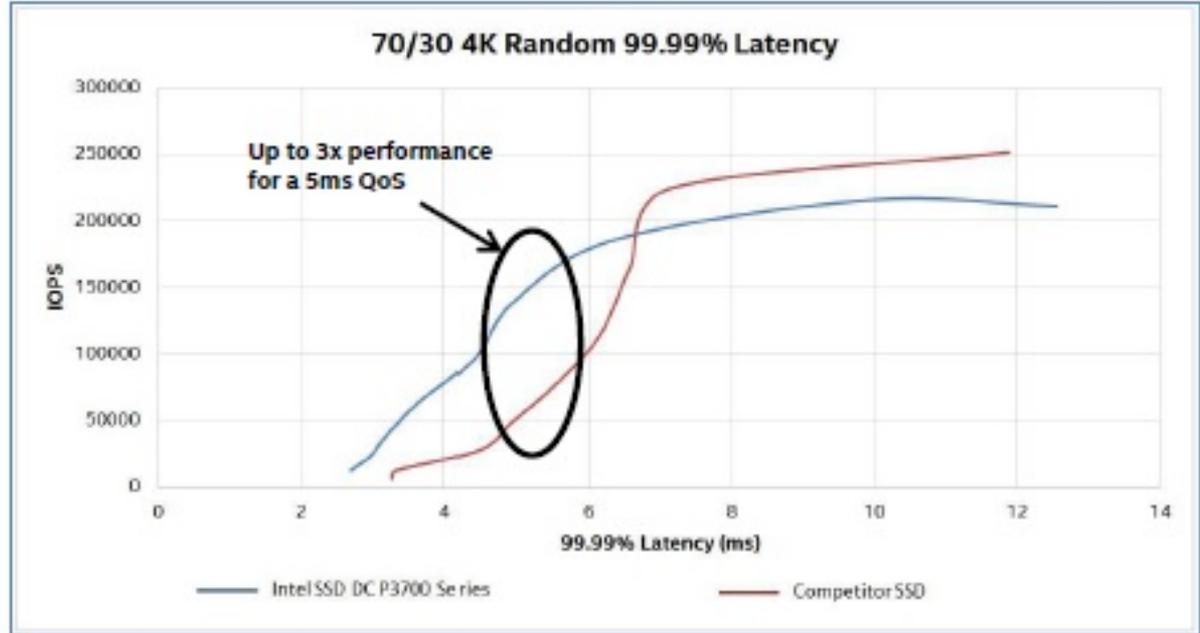
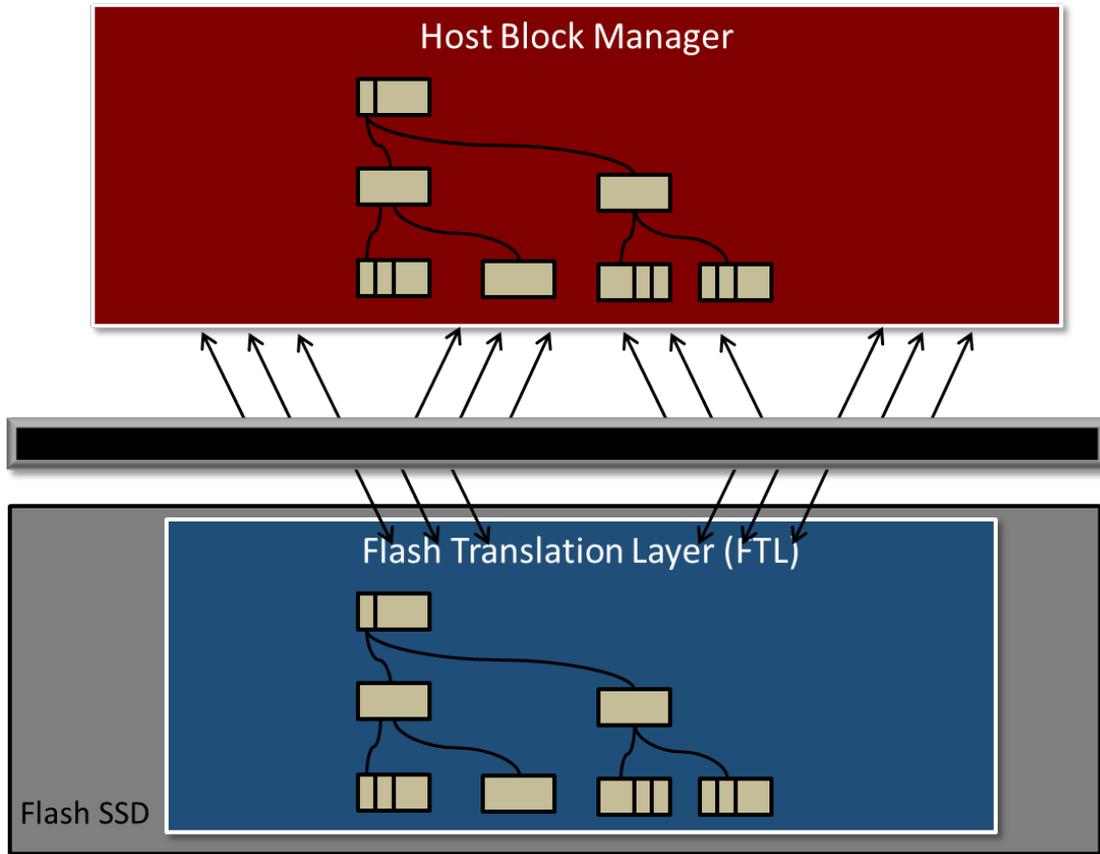
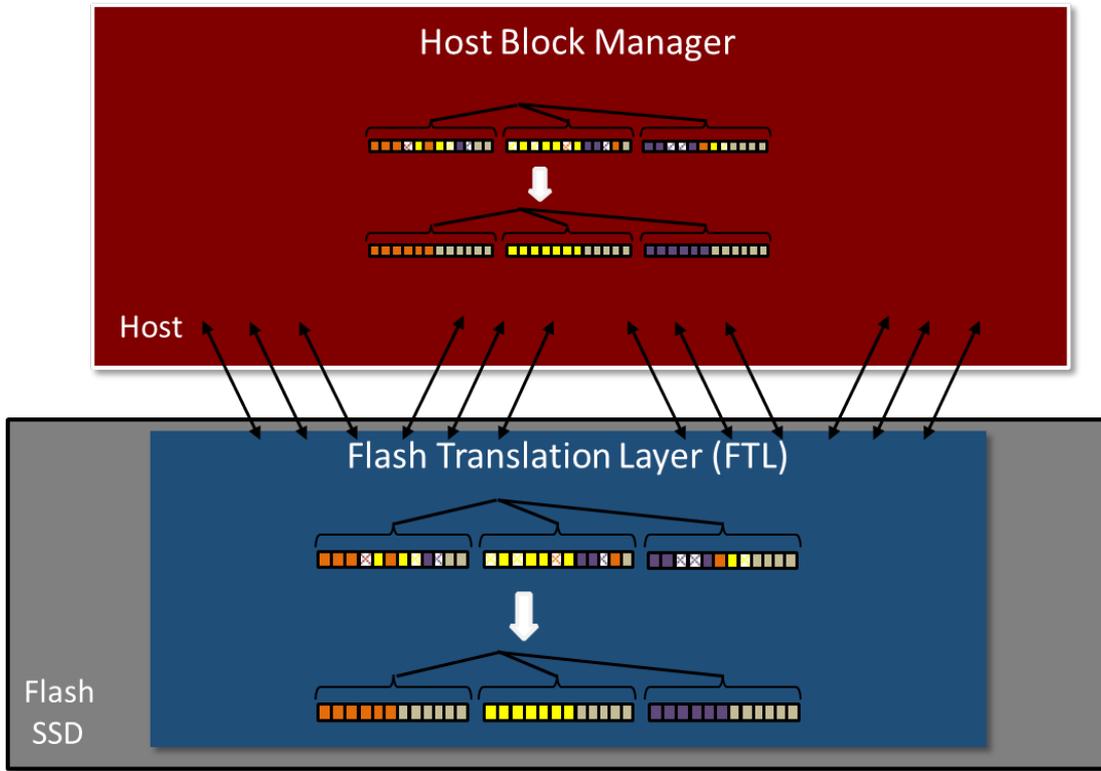


Figure 3: Relationship between 99.99% QoS and Performance (Source: Intel)

Latency  
@  
IOPS







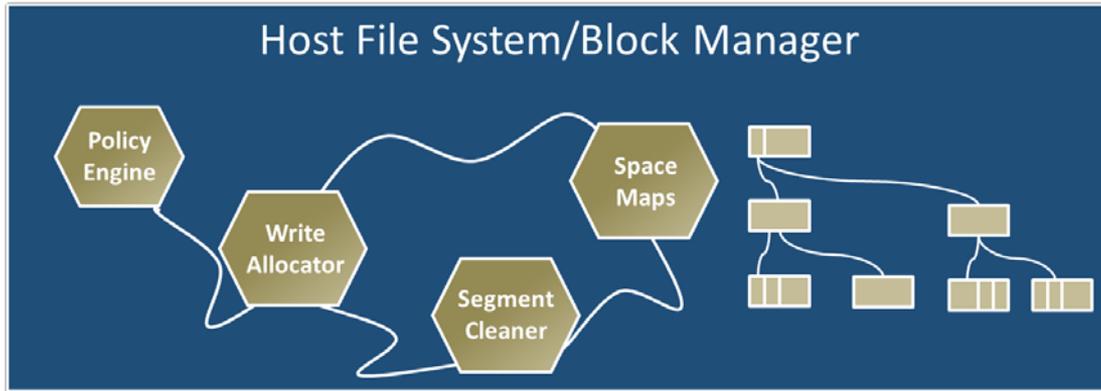
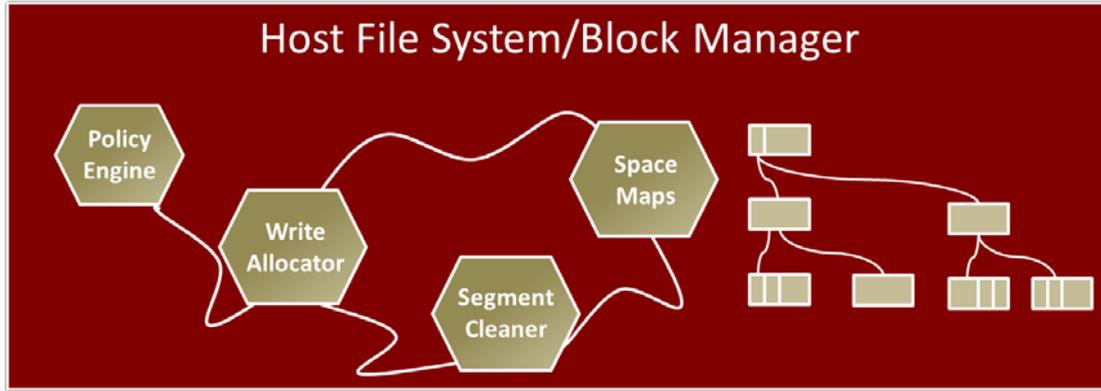
# Standardizing Storage Intelligence for SSDs

- TRIM
- Hinting and Optimal Stripe Size

## FTL SSD Intelligence

- Multi-Stream Writes
- Intelligent placement of data on the storage device
- Intelligent management of garbage collection and overprovisioning

# Challenges with FTL Storage Intelligence



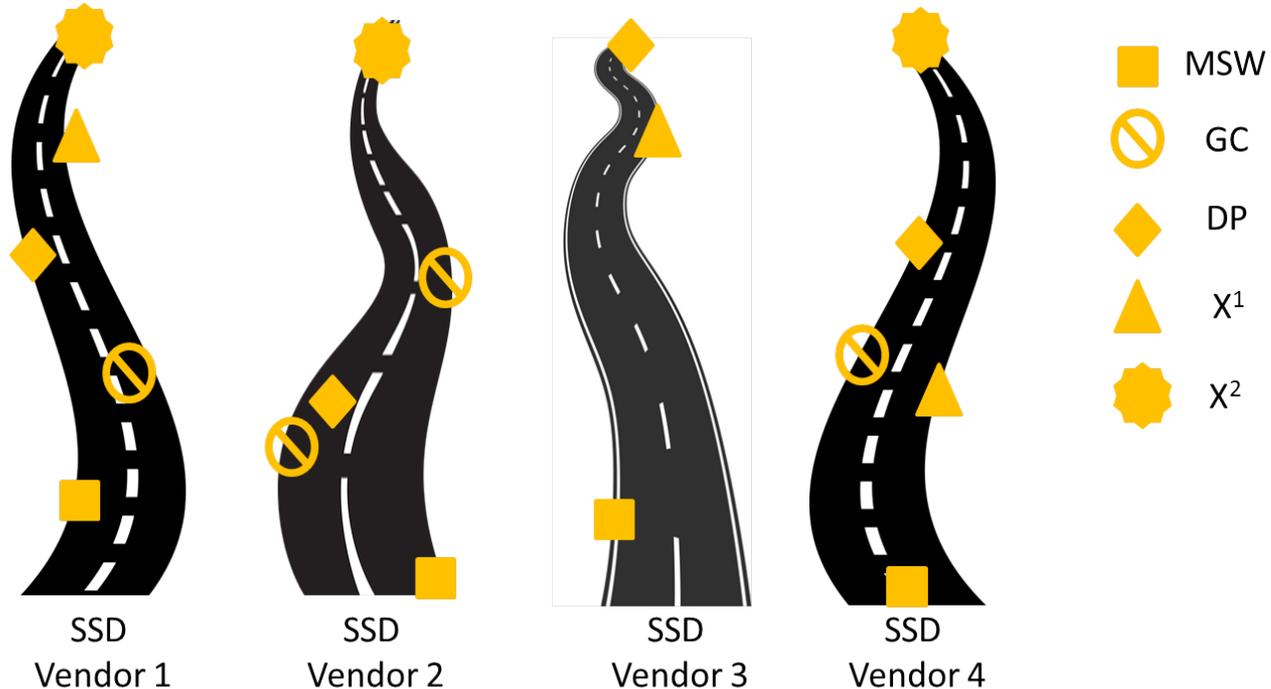
# Challenges with FTL Storage Intelligence

## Universal API

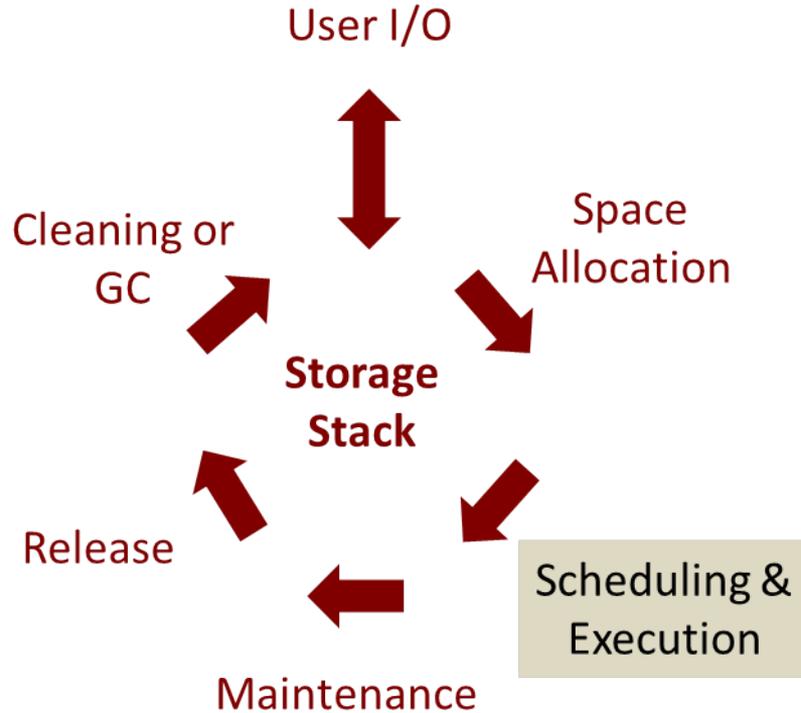


- Huge number of potential parameters
- Information on caching, prioritization, garbage collection, scrubbing, hot/cold
- Information has to be continuously prepared and communicated on every op
- Set up overhead on system software and through communication chain
- Complex Host/Device Contract

- When and what function will be offered by each vendor?
- Will need to continuously add options
- Each vendor's implementation will produce different results, interoperability, and dependencies

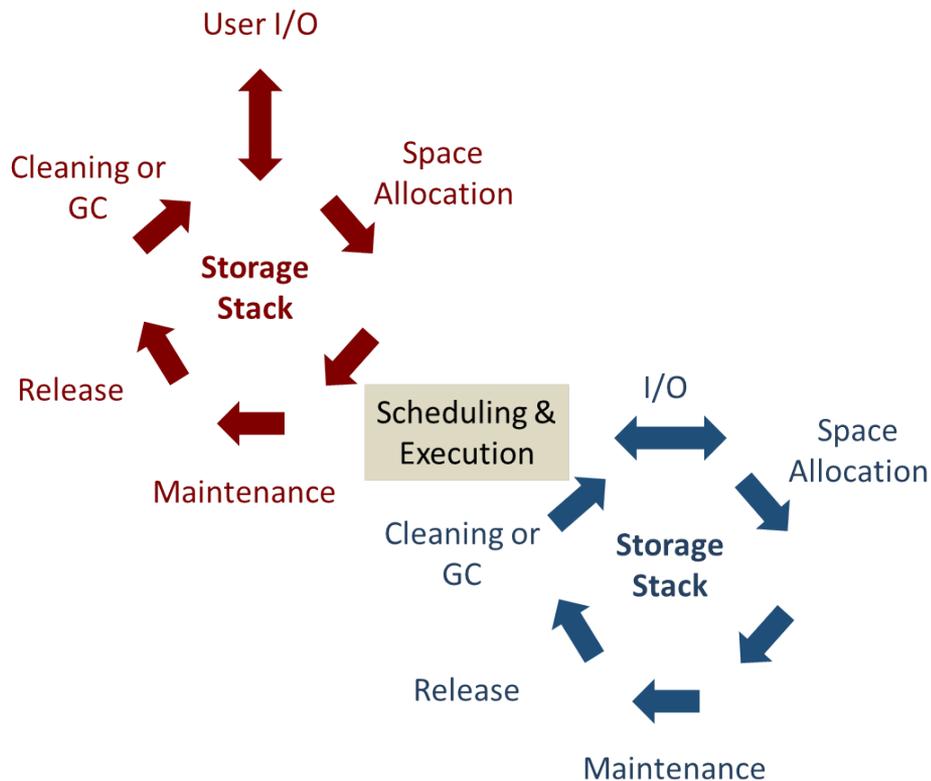


# Data Lifecycle in Storage Software



- Management is required across the life cycle of data
- Process alignment

# Data Lifecycle in Storage Software



- FTLs duplicate these processes
- High probability for disconnects
- Performance and latency impacted with every disconnect

**Space Mgt.**



**Segment Cleaner**



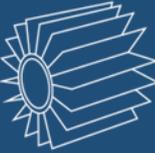
**Scheduler**



**Space Mgt.**



**Segment Cleaner**



**Scheduler**



Space  
Mgt.



Segment  
Cleaner

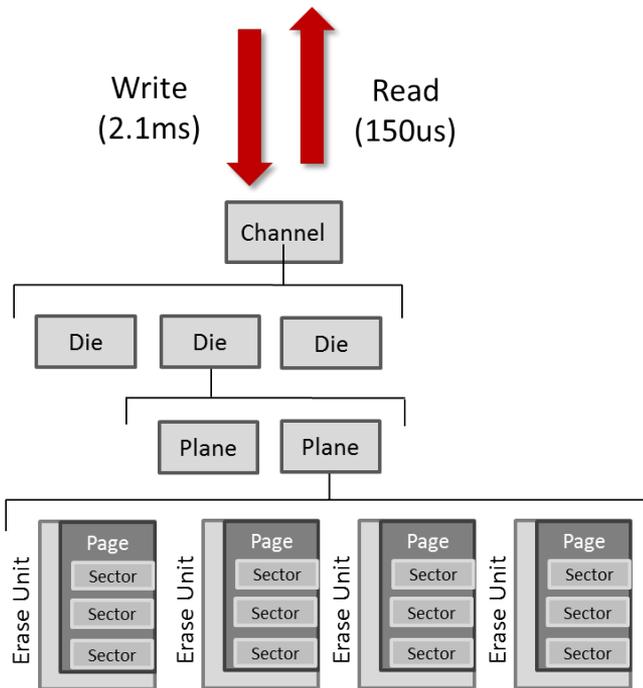


Scheduler



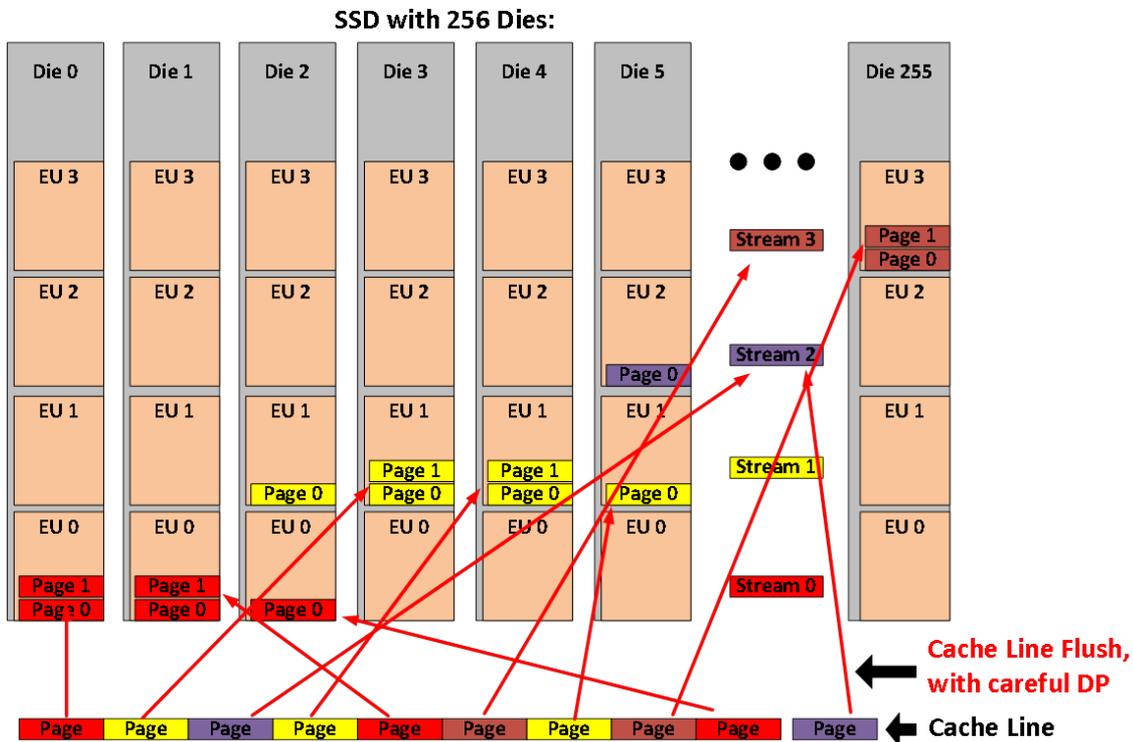
# SSD Storage Intelligence and Latency Spikes

## Read collisions

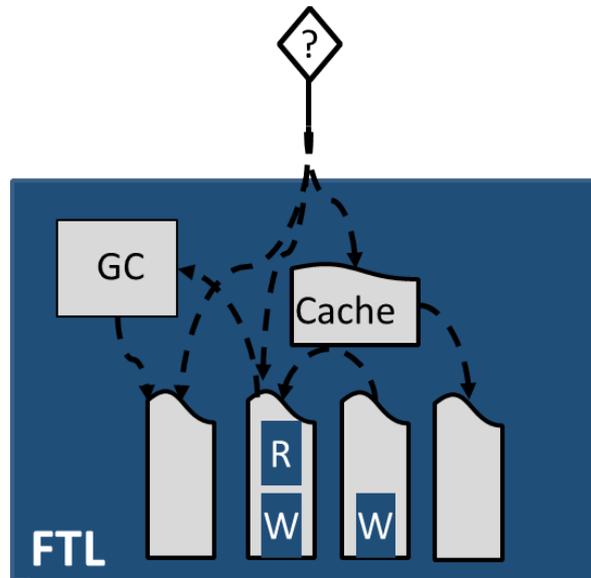


*“Your Read latency  
is  
your Write latency!”*

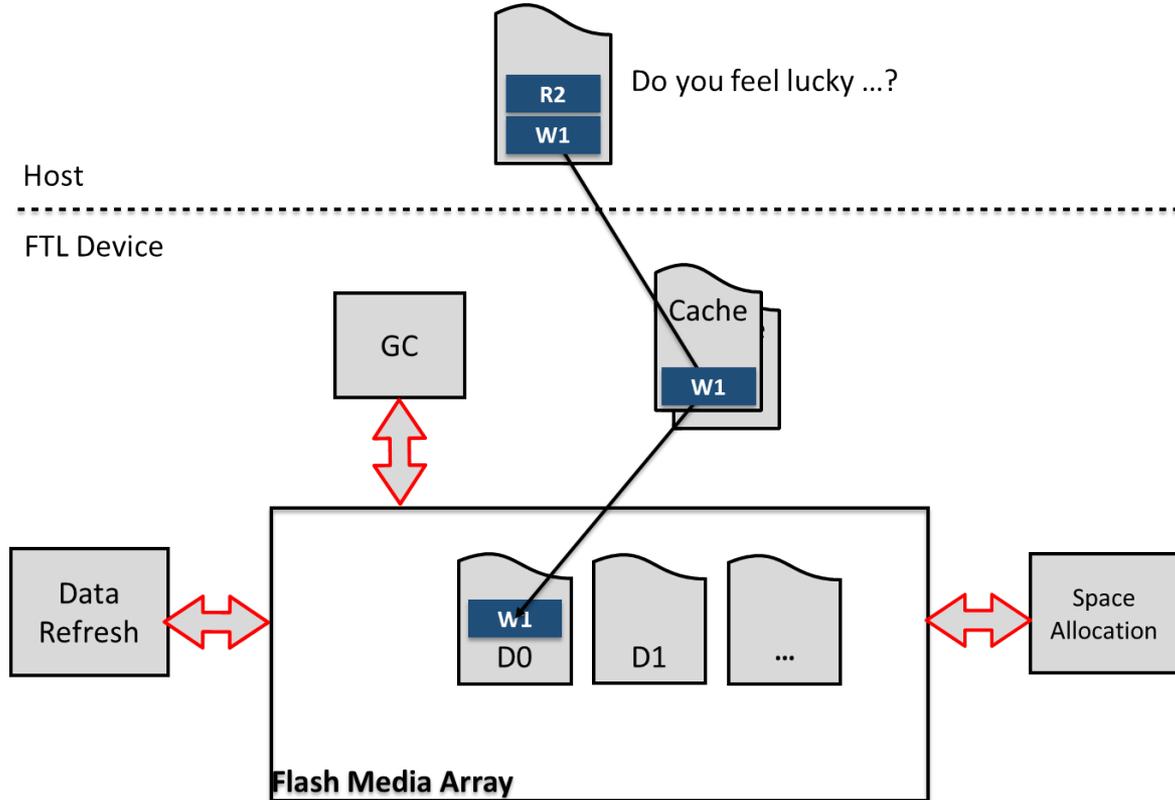
# SSD Storage Intelligence and Latency Spikes



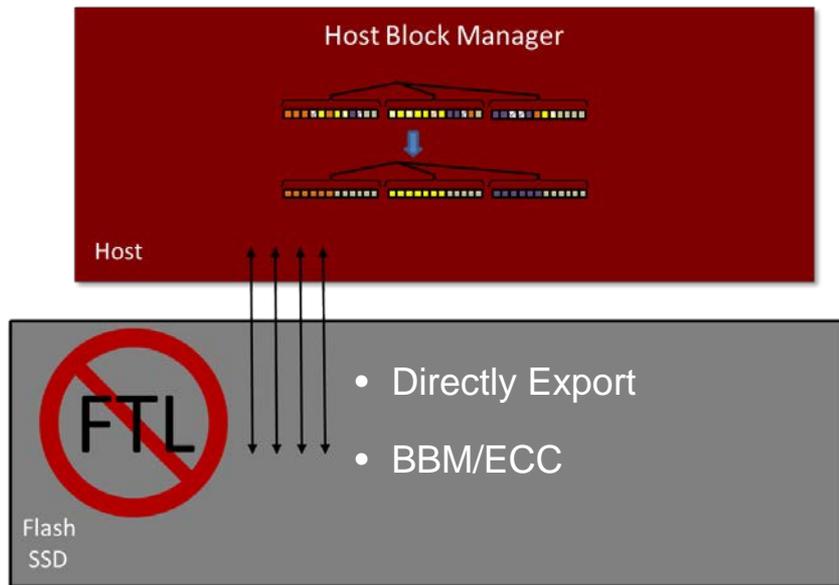
- Proprietary I/O pipeline
- Internal system with async processes: cache flush, garbage collection, data refresh, etc.
- Complex controls to mitigate



# Lifecycle Scheduling and Execution



# Software-Defined Flash



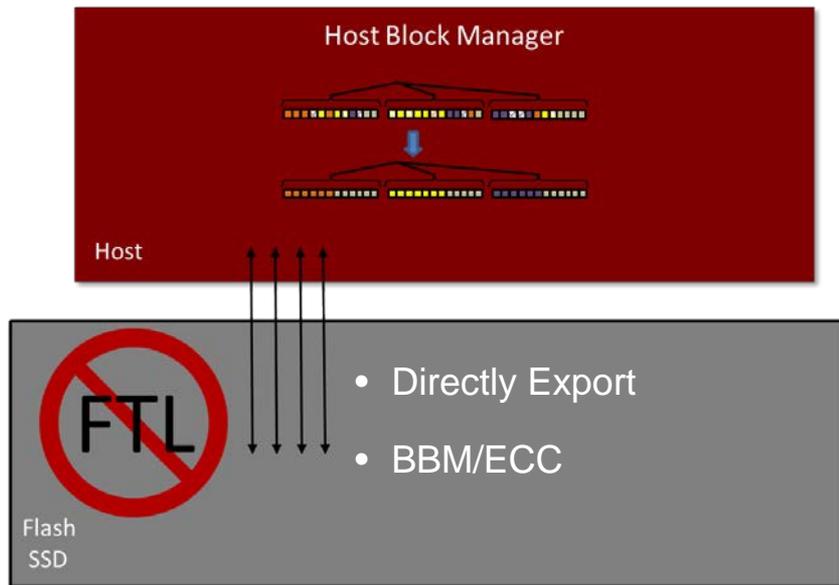
## Software-Defined Flash:

- No FTL
- Direct Geometry Export
- Host stack performs all Flash Management processes

## Results:

- Enables maximum parallelization
- Dramatic performance improvements

# Software-Defined Flash



## Challenges:

- Integration
- Burdens system with NAND attributes and constraints
- Does not provide Forward Compatibility
- Poor RAS capabilities
- Poor Scalability

Flash is now the most important  
Storage System media

But every Flash SSD's  
software is designed to  
emulate a  
Hard Disk Drive



**Flash SSD**

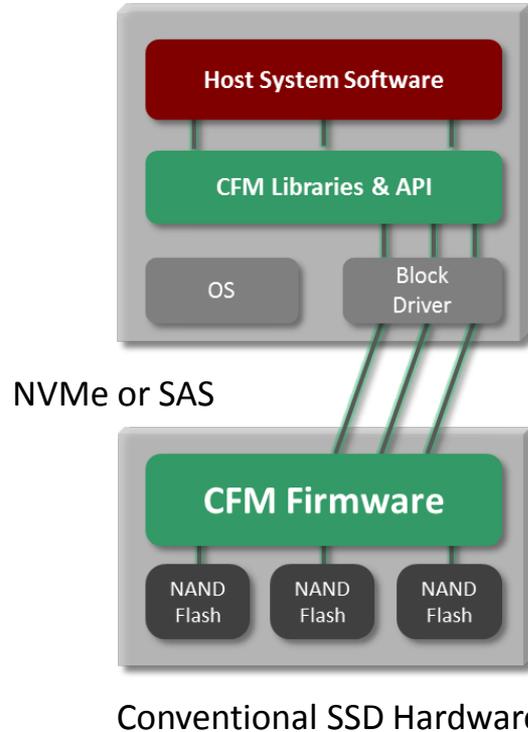


**Hard Disk  
Drive**

# Flash is now the most important Storage System media

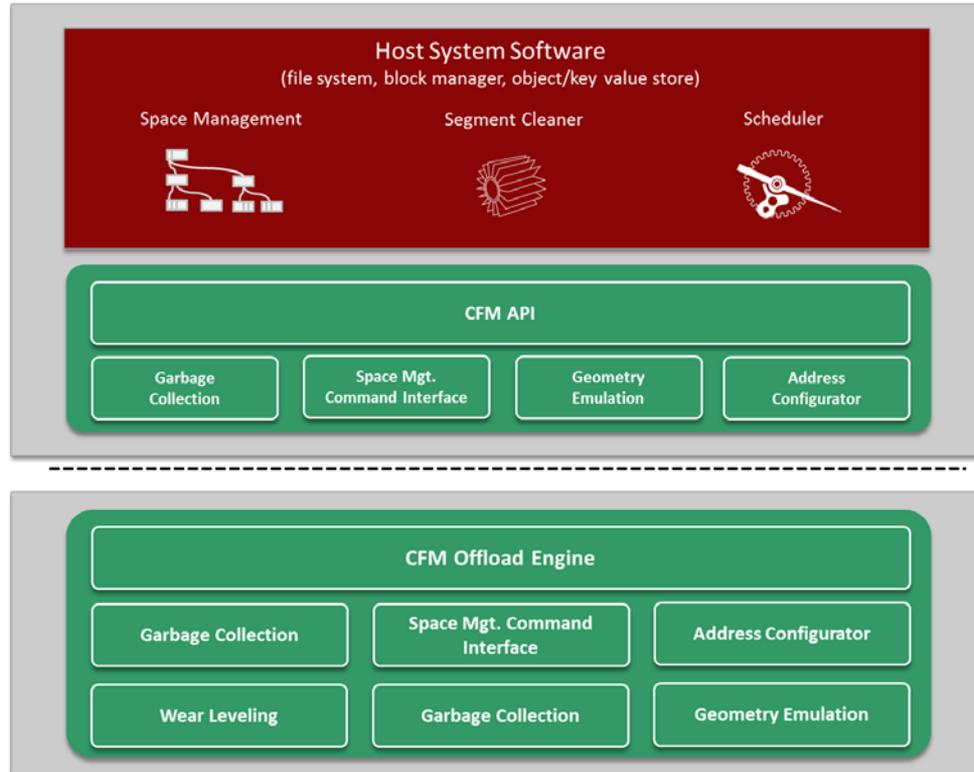


Flash SSD

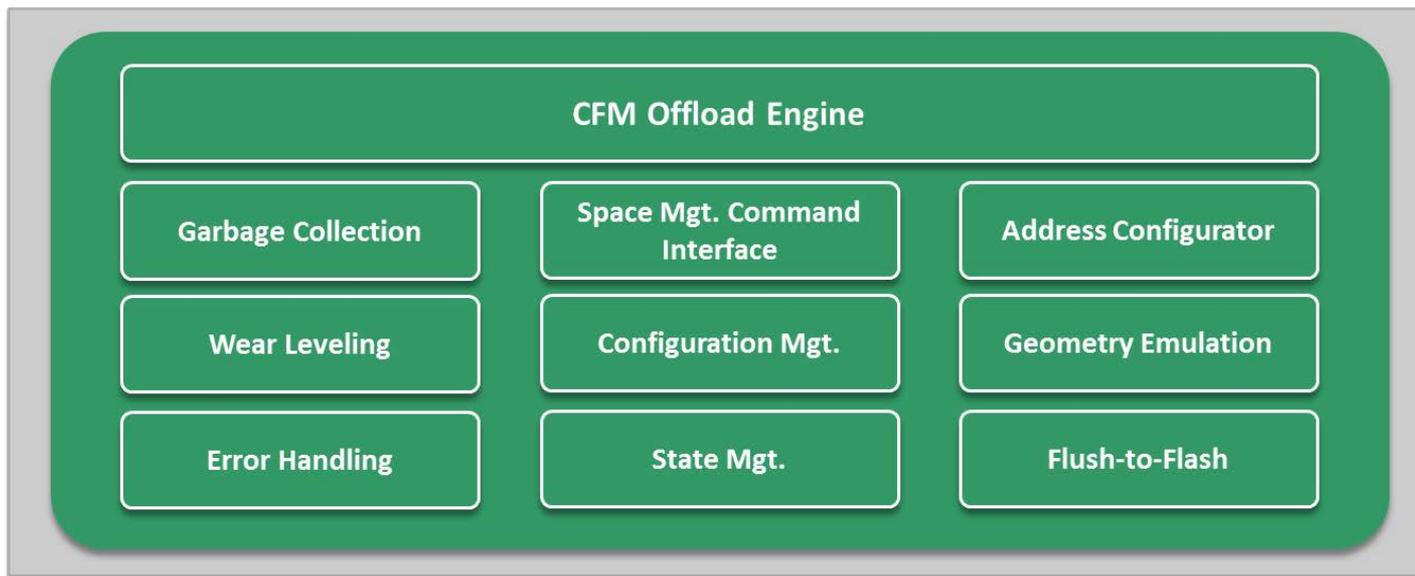


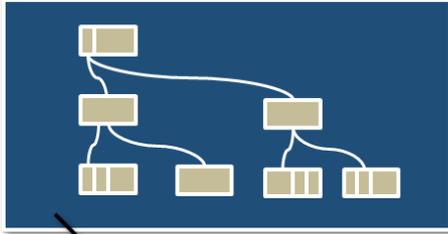
**Cooperative  
Flash  
Management**

# System-Driven Architecture

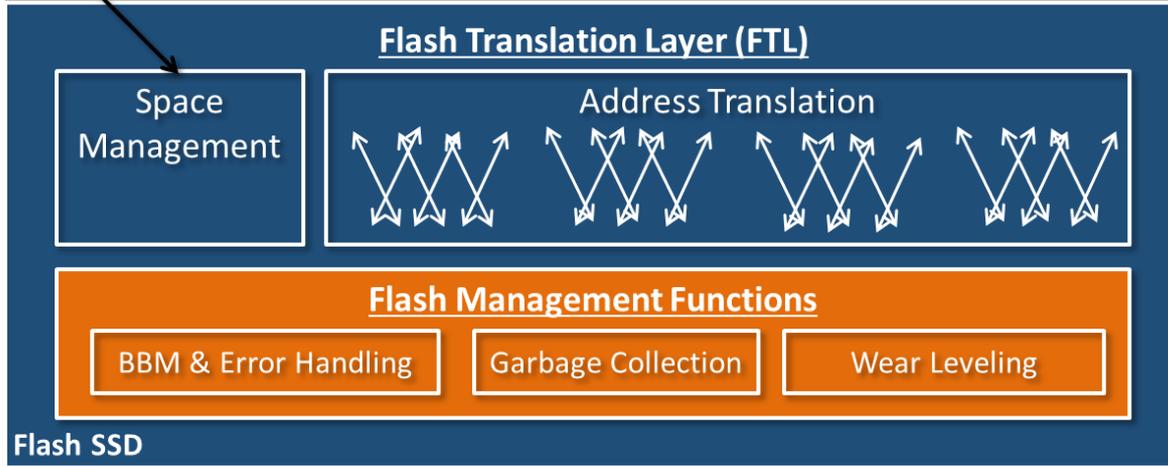


# Offload Engine

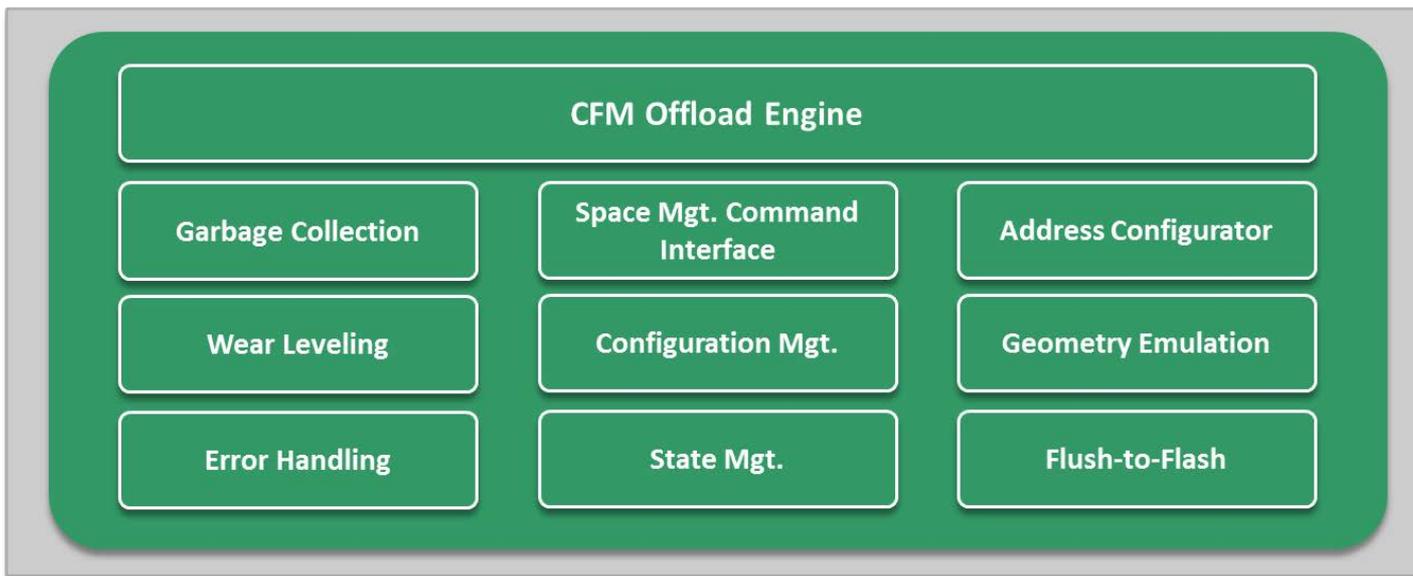




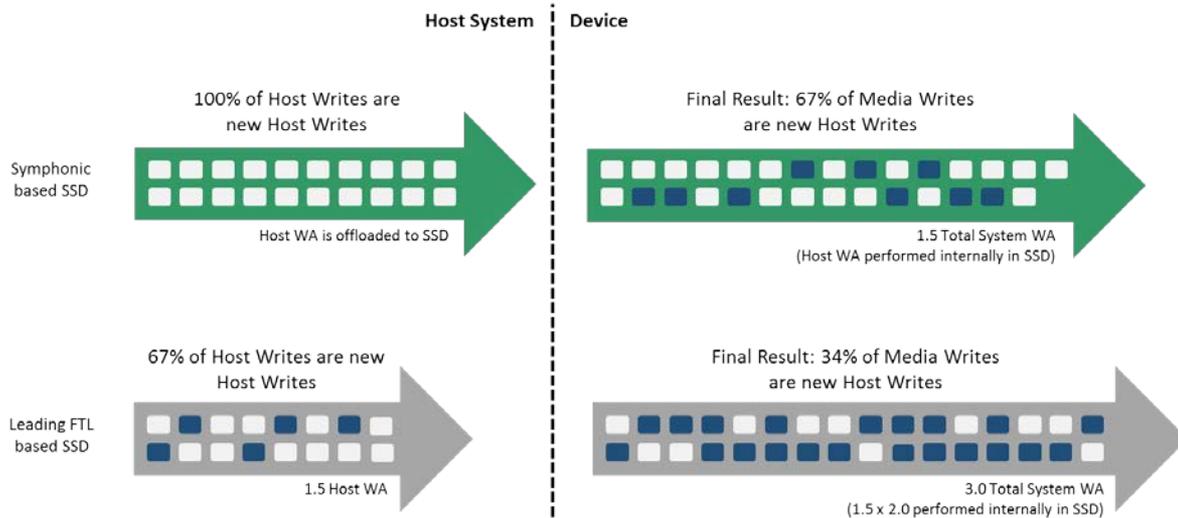
- Backwards Compatible
- Simple Contract



# Offload Engine



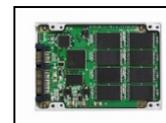
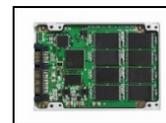
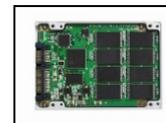
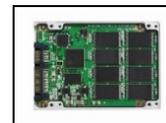
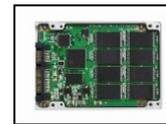
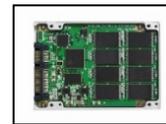
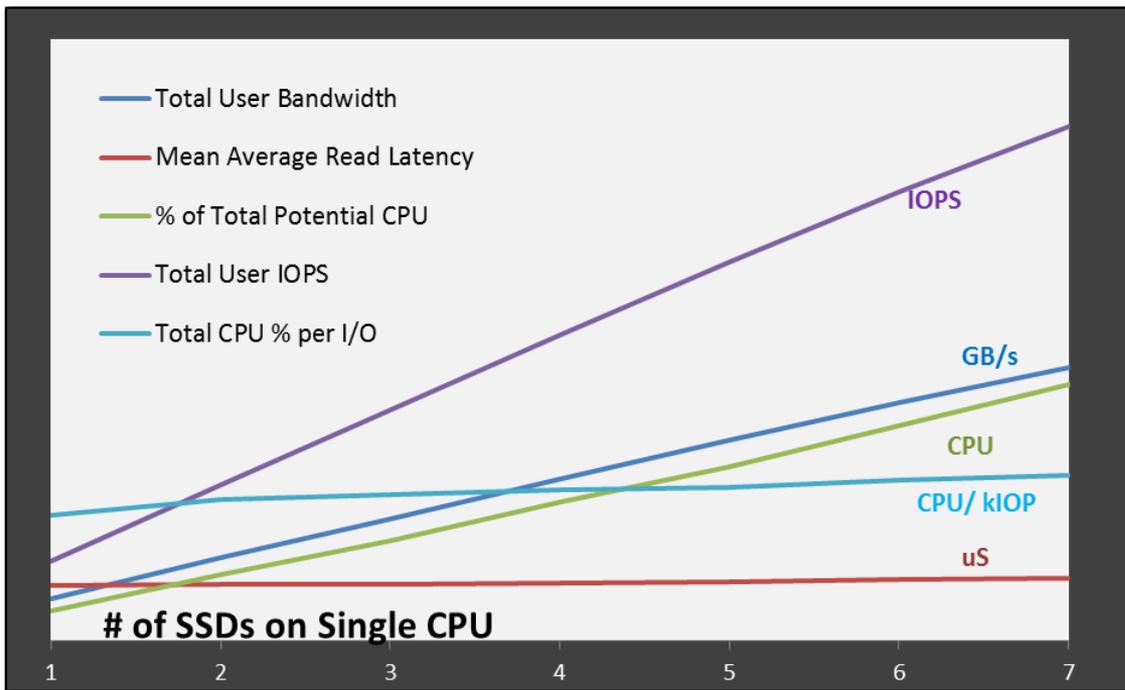
# Offload Engine



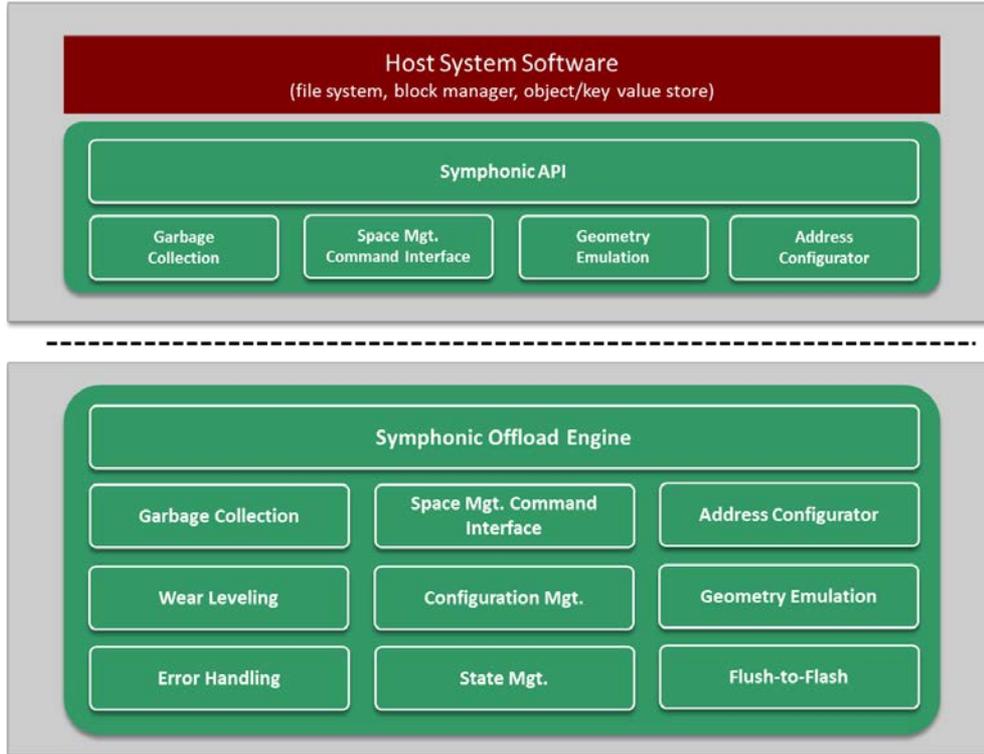
Assumes 1.5 Host WA factor  
 Assumes 2.0 FTL SSD internal WA factor  
 ■ = Copied Data

75% improvement in System-Level Write Amplification

# Scalability

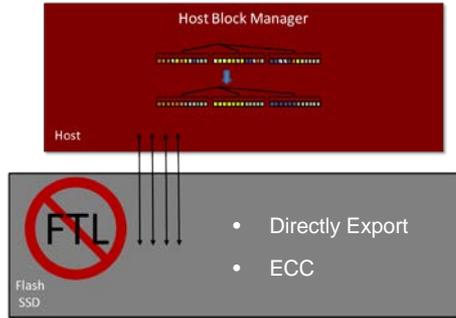


# Abstraction



- Geometry Emulation
- Address Configurator
  - Enables Forward Compatibility
  - Reliability and FRU capabilities
  - Vendor Supported Warranties

# Solving Software-Defined Flash for the Data Center



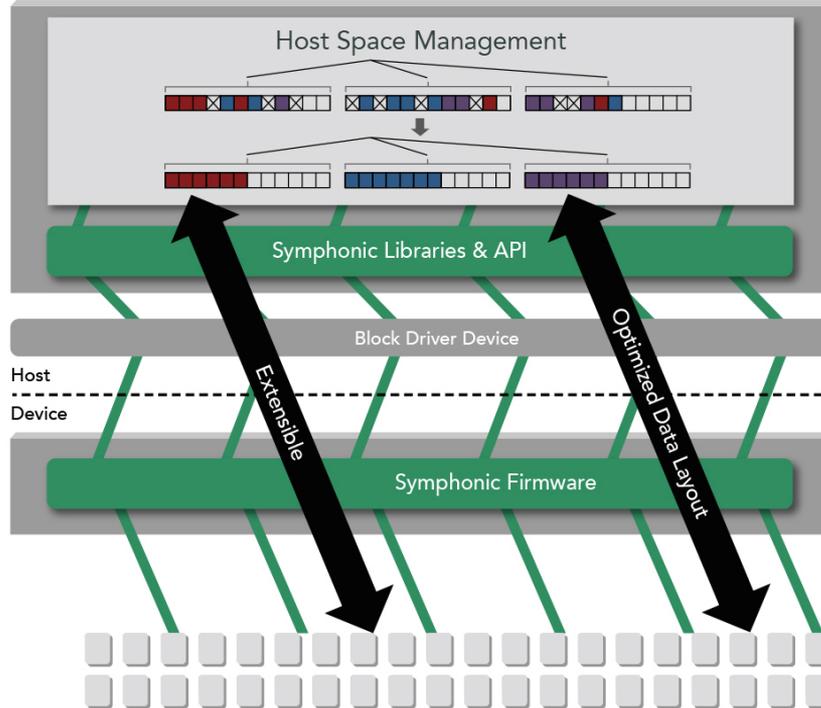
## SDF Challenges:

- Integration
- Burdens system
- Forward Compatibility
- RAS
- Scalability

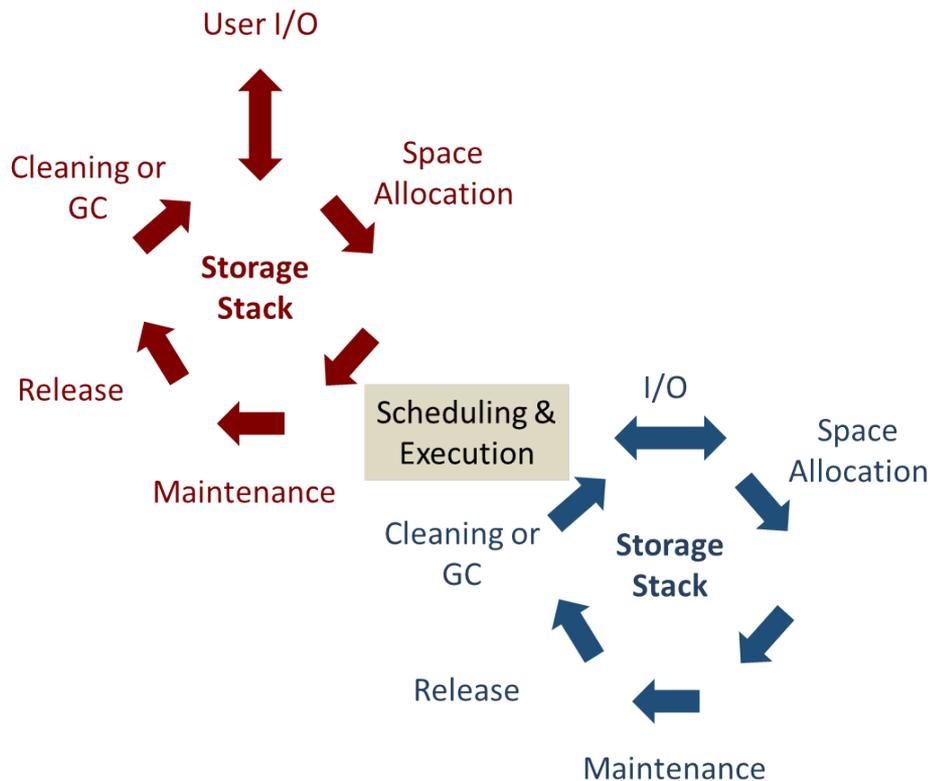
## CFM:

- Geometry Emulation & Address Configurator
- Offload Engine
- Wear Leveling

# Extensibility & Parallelization

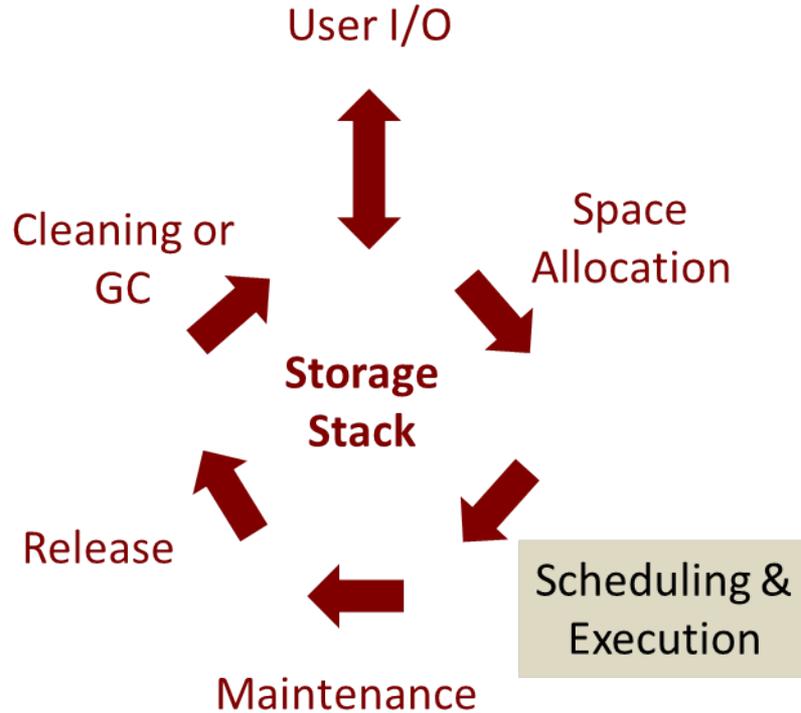


# Data Lifecycle in Storage Software



- FTLs duplicate these processes
- High probability for disconnects
- Performance and latency impacted with every disconnect

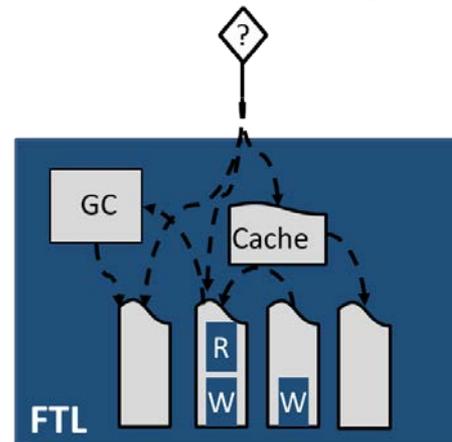
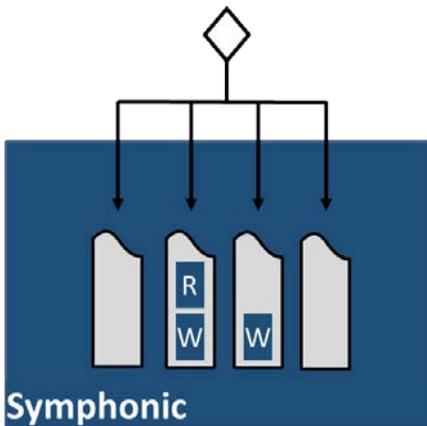
# Data Lifecycle in Storage Software



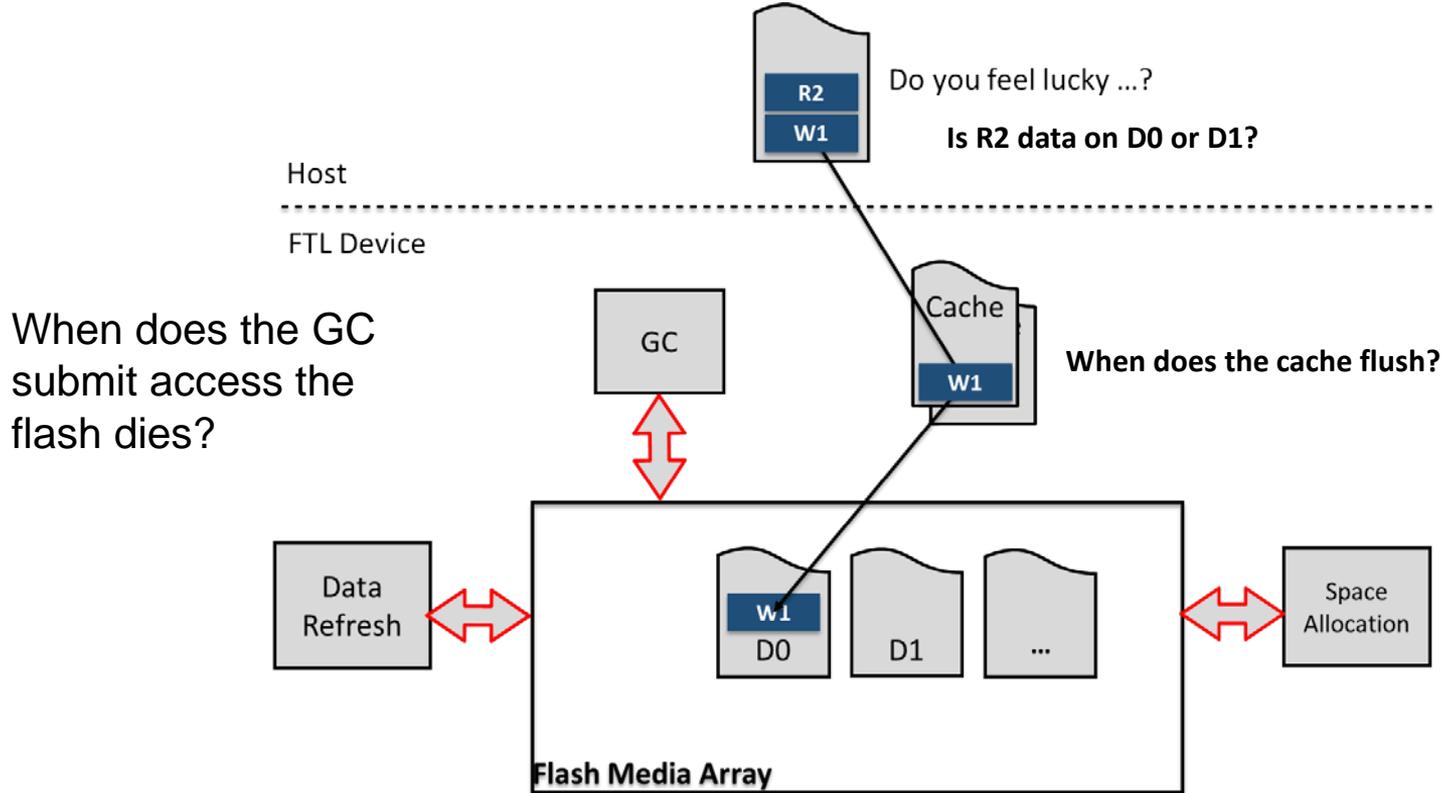
- CFM enables host ownership of the complete lifecycle in a combined system
- Implicitly supports different storage stacks with unique implementations and requirements

# Lifecycle: Scheduling and Execution

- CFM enables host optimization of flash resources
  - High Visibility into I/O queuing
  - High Parallelization
  - Predictable operation
  - Low-Latency & Low-Jitter
- I/O pipeline proprietary
- Internal storage system
- Internal Async processes (cache flush, gc, data refresh, etc)
- Complex controls to mitigate

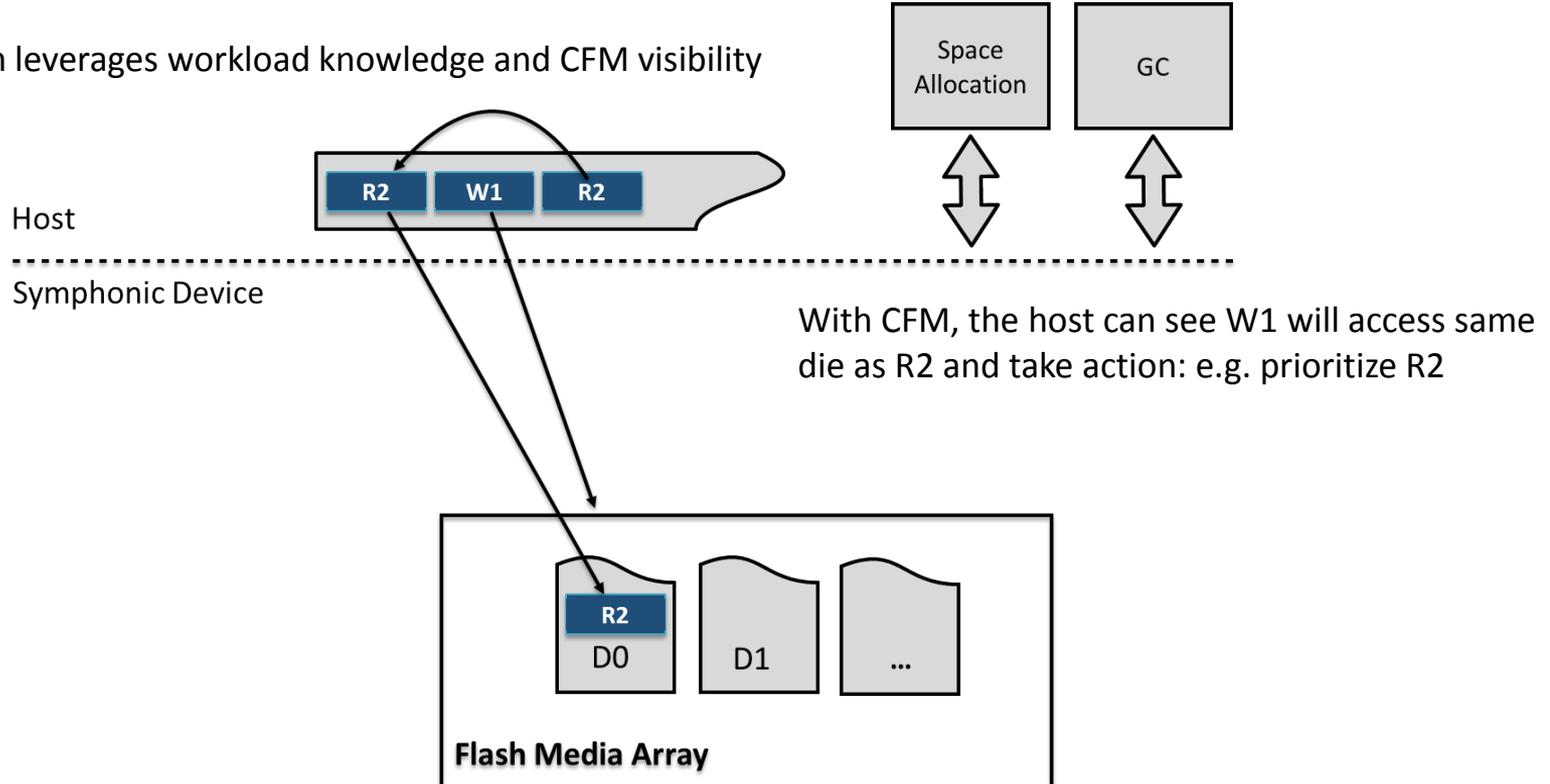


# Lifecycle: Scheduling and Execution



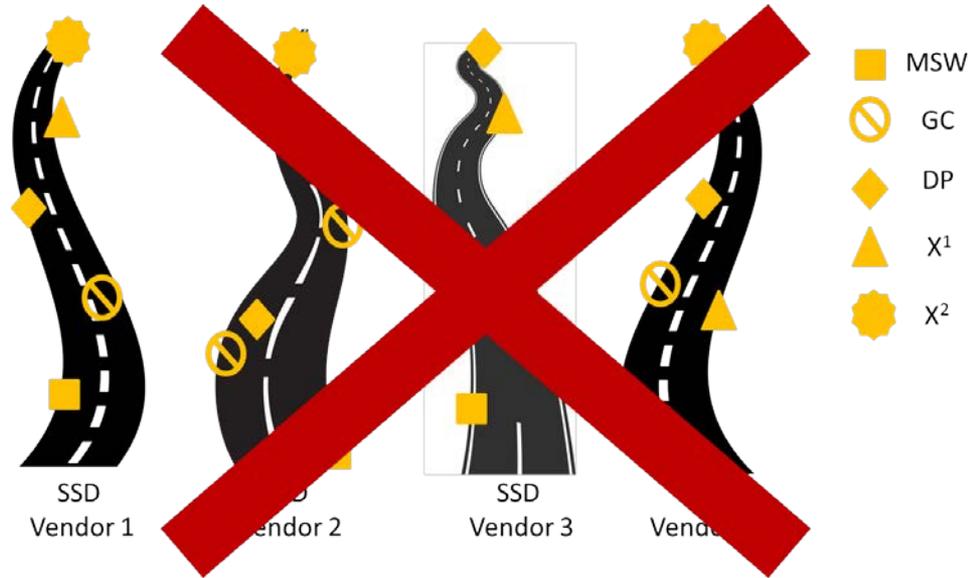
# Lifecycle: Scheduling and Execution

System leverages workload knowledge and CFM visibility



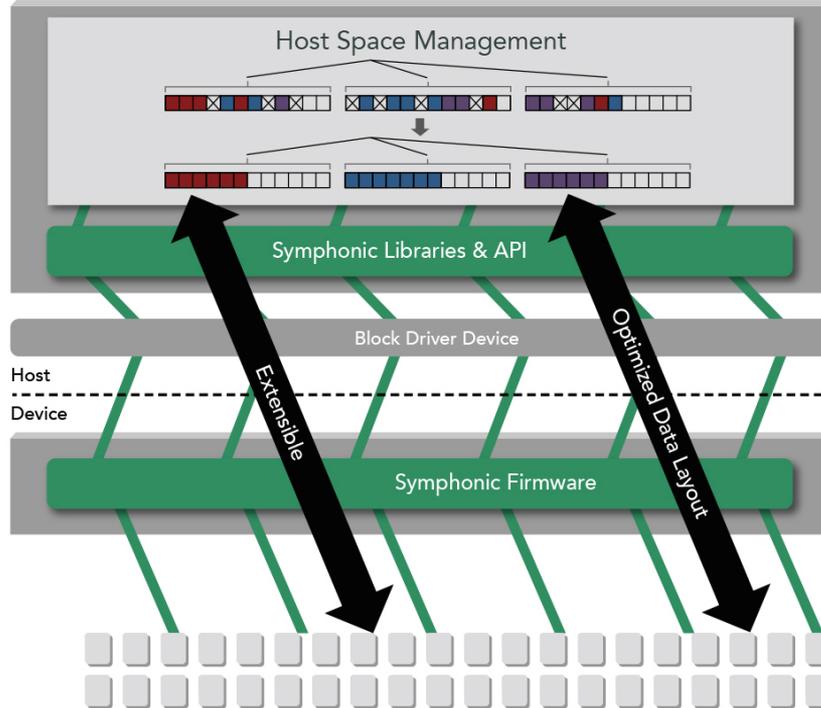
# CFM Host Control

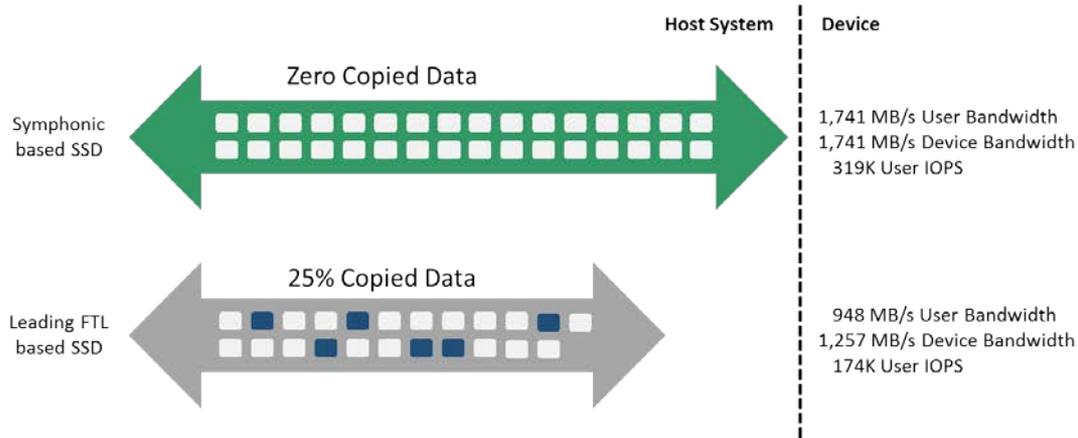
- System intelligence is left in the host and preserved to the media, so it's independent of the SSD



- Simple host contract from the outset – not a complex evolving interface

# Extensibility & Parallelization

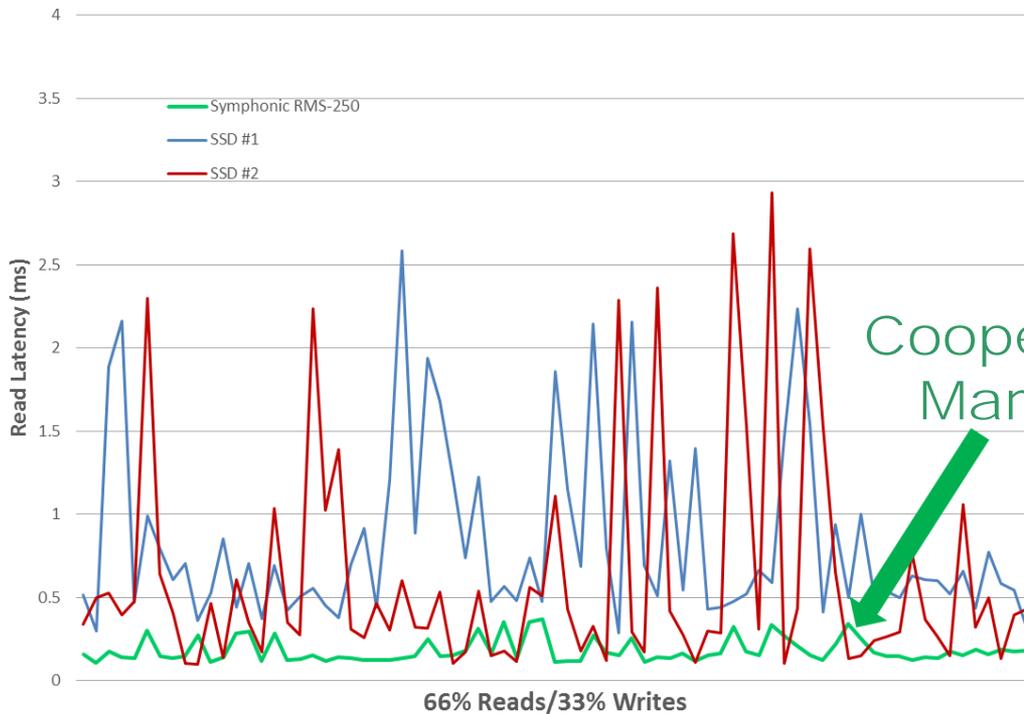




Workload: 66% random reads/33% sequential writes on random 8MB segments  
Device Queue Depth: 128; Assumes 1.5 Host WA factor

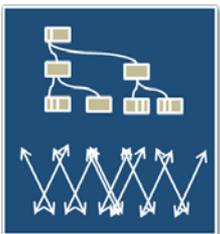
## >80% improvement in IOPS and Bandwidth

Symphonic firmware turns the SSD into an offload engine, eliminating the copying that would normally occur between the host and SSD (copy overhead that would even occur with a HDD).

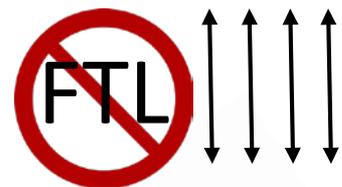


Cooperative Flash Management

# Extremes

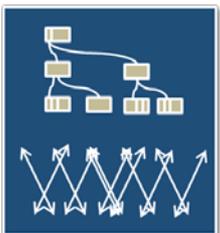


**FTL SSD**

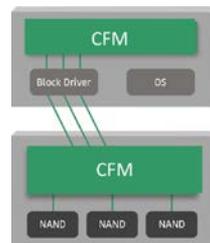


**True  
SDF**

# Goldilocks



**FTL SSD**



**CFM SSD**



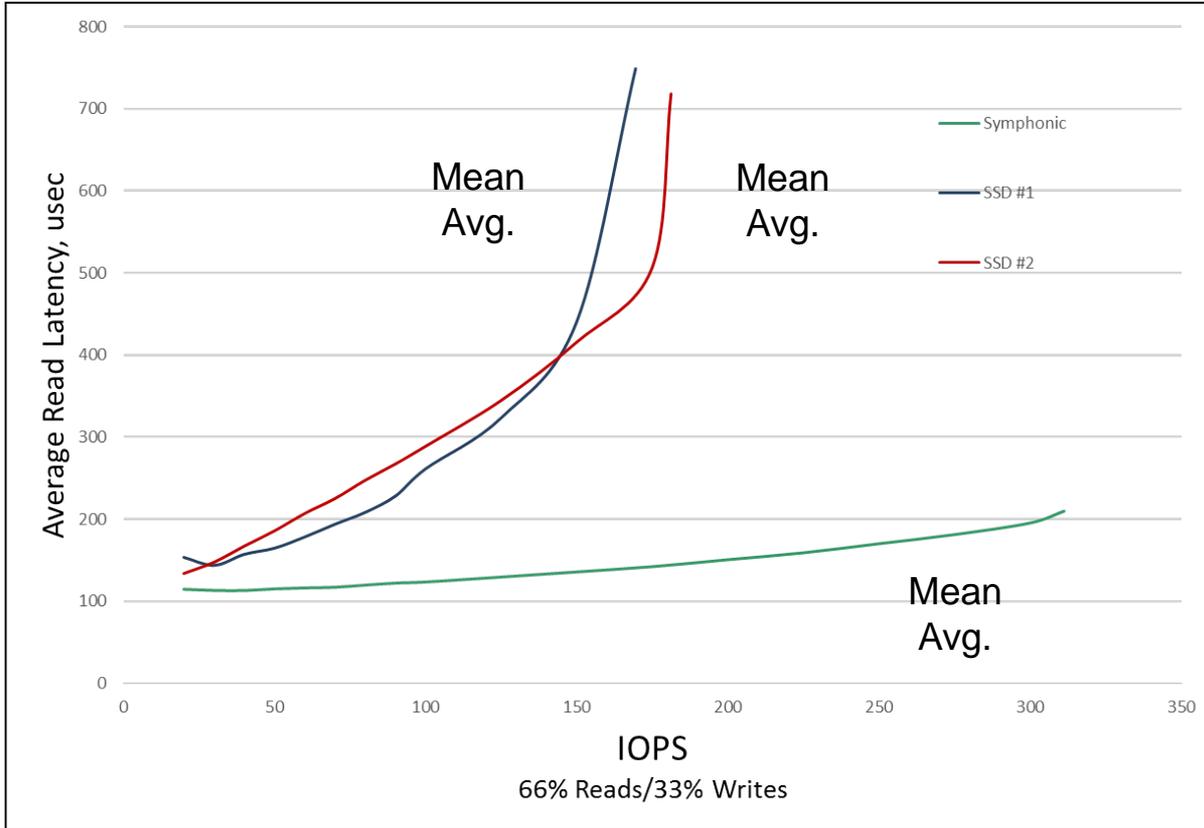
**True SDF**

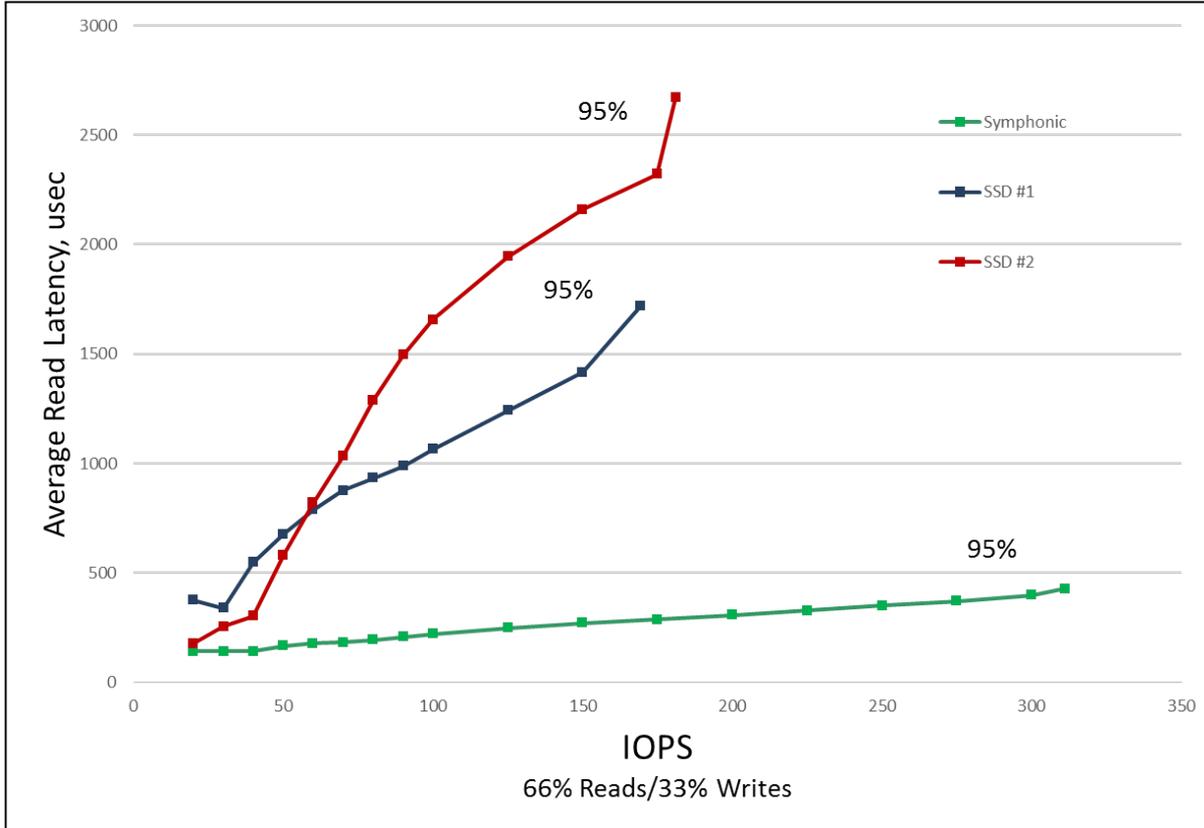
Flash is now the most important  
Storage System media

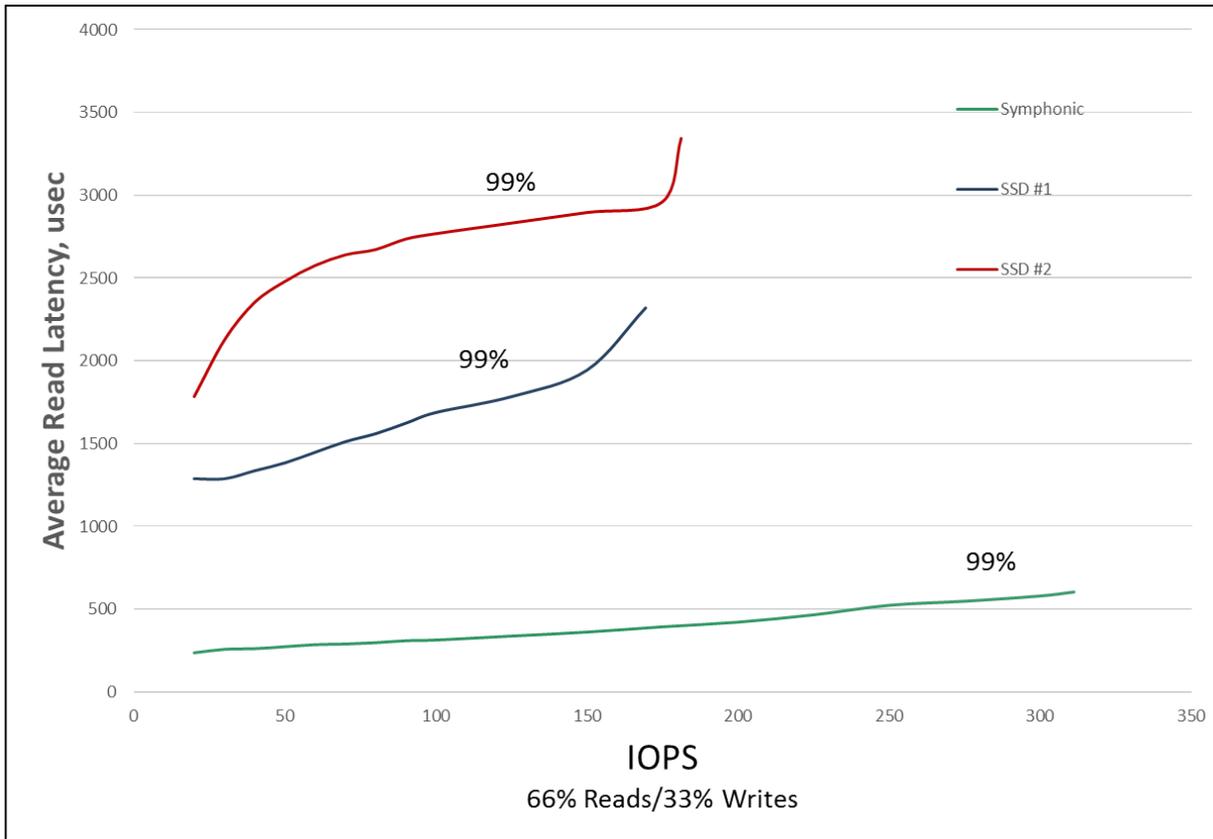
2015

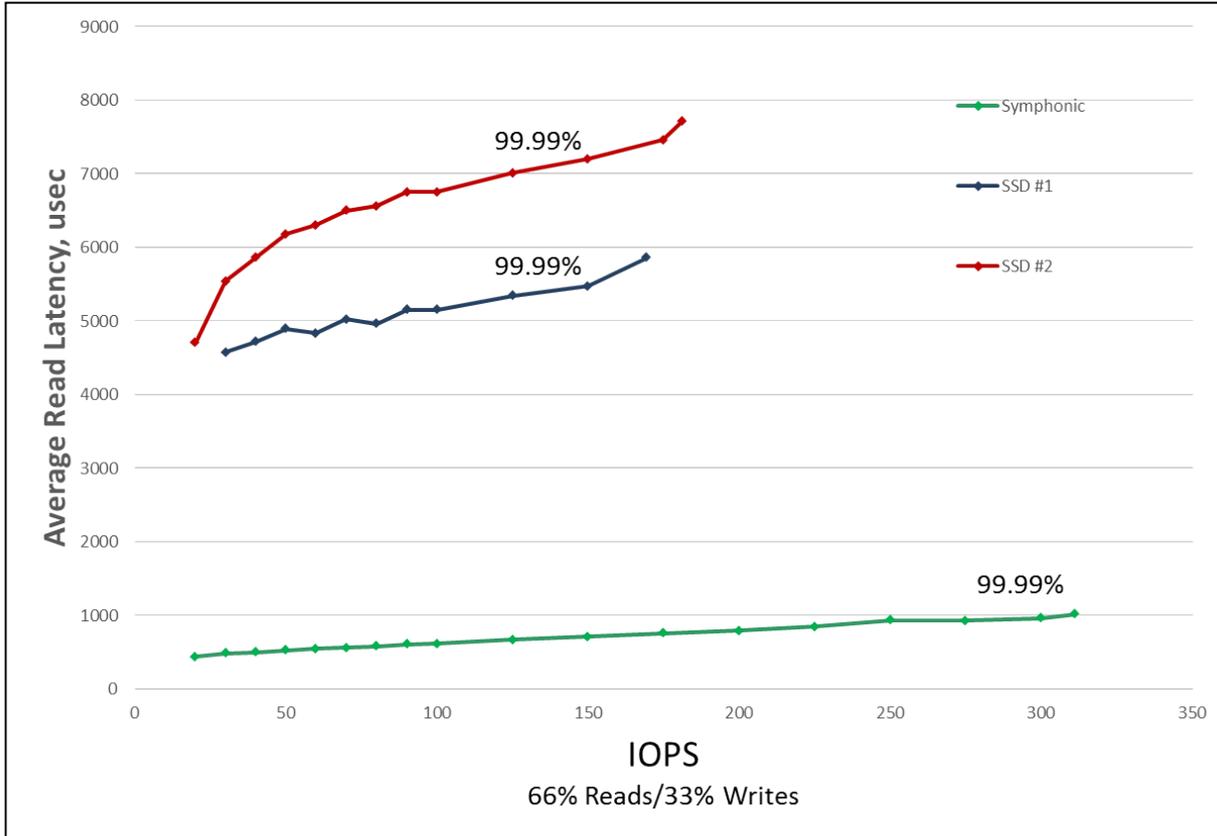


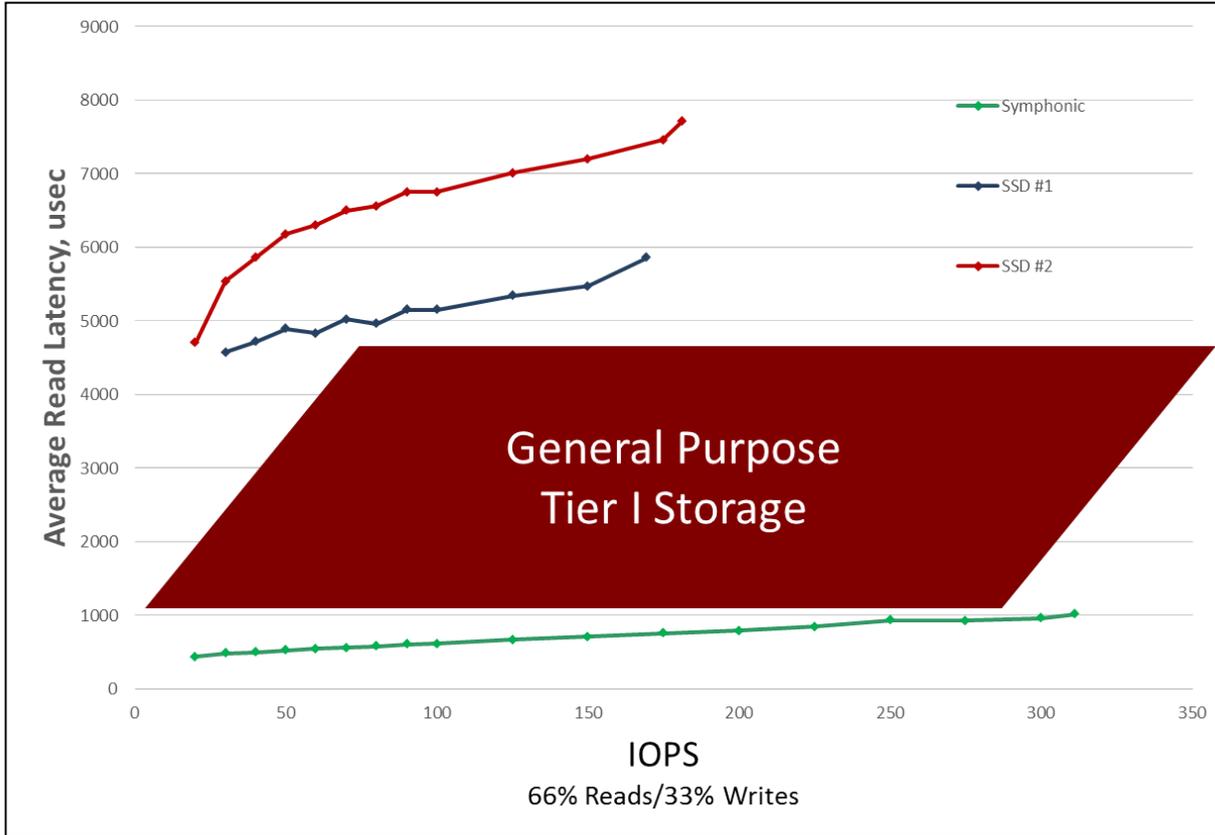
Where does Flash go  
from here?











# Data Center Primary Storage



WINTER IS COMING



HDD



Flash



DRAM

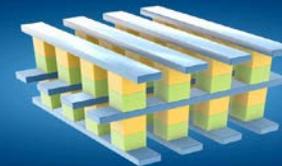
# Data Center Primary Storage

WINTER IS COMING

HDD

Flash

3D XPOINT™



DRAM

# Data Center Primary Storage

WORM

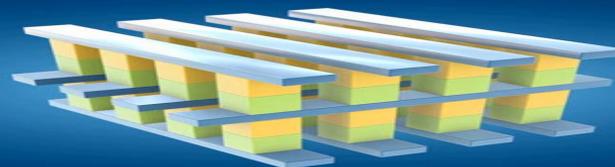
General  
Purpose

RDBMS

HDD

FTL  
Flash

3D XPOINT™



DRAM

# Data Center Primary Storage

